

نسخه إلكترونيه مجانيه

مبادئ في لغة ال C++

بقلم :
معاويه نورالدين

2013



الفهرس

مقدمه

الحمد لله ملء السموات و الأرض ، وملء ما بينهما ، وملء ما شاء الله من شئ بعد .
والصلاة والسلام علي عبد الله المصطفى وعلى نبي الله المرتضي وعلي رسول الله المجتبى وعلى آله وأصحابه ومن بهم إهتدى. (سبحانك لا علم لنا إلا ما علمتنا إنك أنت العليم الحكيم)

أقدم لكم هذه الصفحات التي تناولت فيها مبادئ للغه الـ ++C بشكل مبسط وشرح مباشر وأرجو أن تكون كذلك ، وتتوافق مع مقرر بعض كليات الحاسوب بالسودان وهي مكمله للنسخه الأولى من هذا العمل وأسال الله العظيم رب العرش العظيم أن ينفعني وإياكم بها وأسالكم الدعاء فإني أحوج ماأكون لدعوه الأخوان . وإذا كان لديك أي إستفسار ، أو ملاحظه فيمكن إرسالها إلي عبر البريد الإلكتروني mawia93@gmail.com وشكراً.

أخيكم في الله
معاويه نورالدين
السودان / الخرطوم

2013-12-01

إهداء

إلي أمي الحبيبه وأبي الحبيب أسأل الله أن يطيل بقائهما ويجعل الجنه مسكنهم

إلي كل من قدم للأمه الإسلاميه من دعاء إلي جهاد في سبيل الله

إلي زملاء الدراسه بكليه المشرق للعلوم والتكنولوجيا

إلي روح جدي أسال الله أن يجعل الجنه مثواه

إلي كل من شجعني إلي كتابه هذا العمل

إلي كل من له فضل علي

العامل الشرطي conditional operator

لا يختلف هذا العامل عن التركيب الشرطي للأداة if…else بل هو إختصار لها بحيث يمكن كتابه الشرط وجوابي الشرط في سطر برمجي واحد وصيغته العامه :

if (condition) ? Statement..1 : Statement..2 ;

كما هو واضح نكتب الشرط بين القوسين ونكتب الجمله البرمجيه المراد تنفيذها في حاله تحقق الشرط مكان Statement..1 ونكتب الجمله البرمجيه المراد تنفيذها في حاله عدم تحقق الشرط مكان Statement..2 .

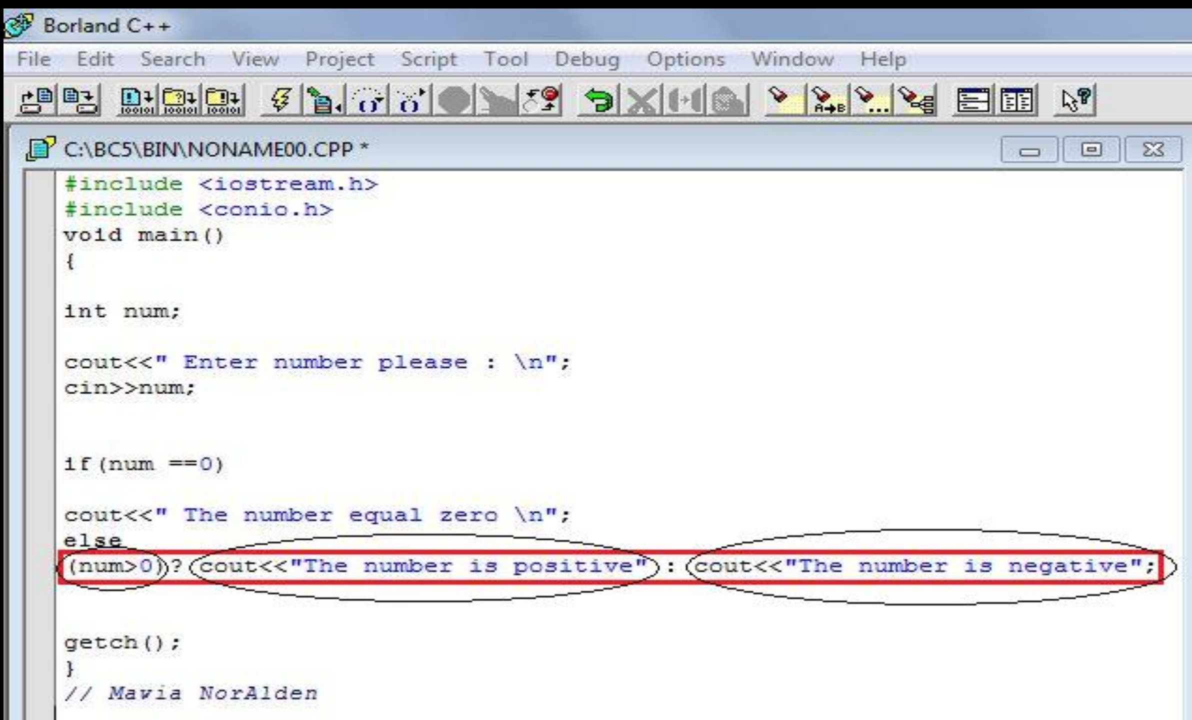
مثال لذلك :

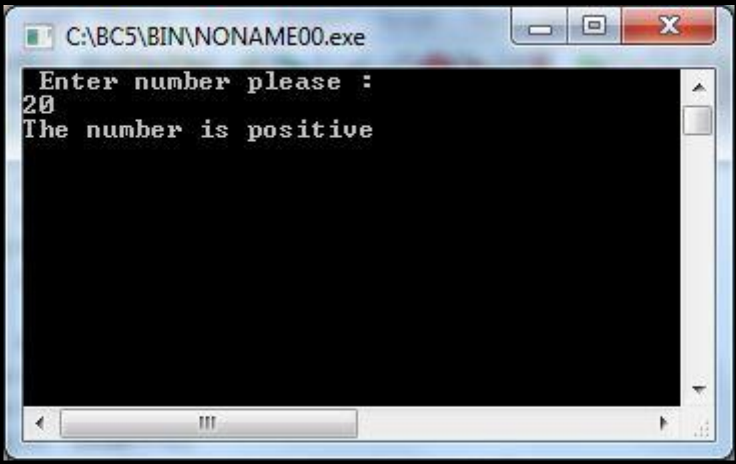
❖ برنامج لإختبار الرقم سالب أم موجب بخلاف الصفر.

مناقشه :

سنقوم بكتابه برنامج يستقبل رقم من المستخدم .
ونختبر هذا الرقم هل يساوي الصفر؟ لأننا لا نريد إختبار الصفر بواسطه العامل الشرطي.
وفي حاله الرقم المدخل يساوي صفر ، يظهر رساله .وإلا يقوم بإختبار الرقم هل هو سالب أم موجب بإستخدام العامل الشرطي ويقوم بطباعه رساله توضح نوع الرقم.
ملحوظه : سنستخدم المترجم Borland c++ لجميع البرامج.

كود البرنامج :





قصدتُ إستخدام التركيبه if… else في نفس الوقت مع العامل الشرطي للتوضيح والفهم .حيث أن جواب الشرط الأول if (num==0) في حاله لم يتحقق . هو شرط أخر له جوابان إما سالب وإما موجب.

يمكننا إستخدام العامل الشرطي وحده من دون إستخدام أداة الشرط if . ويمكن أن نُسند قيمه لمتغير بإستخدام عامل الشرط ومثال لذلك :

```
int y = ( num > 0 )? 5 : 10 ;
```

- ✓ وهذا يعني أن قيمه المتغير y ستصبح 5 في حاله أن المتغير num قيمته أكبر من صفر وإلا ستصبح 10.
- ✓ يمكن أن يكون أحد جواب الشرط هو شرط أخر له جوابان وهكذا.و العامل الشرطي قابل للتكيف مع مثل هذه الحاله .
- ✓ يستخدم العامل الشرطي للتقليل والتسهيل في عمليه كتابه البرامج.

السر الحقيقي وراء النجاح هو الحماس

++ عامل الزياده و النقصان --

نكتب عامل الزياده في لغه C++ بالرمز ++ وعامل النقصان بالرمز -- ونستعملهما مع أسماء المتغيرات ، إما لزياده قيمه المتغير بمقدار واحد ، أو لإنقاص قيمته بمقدار واحد.

ونكتب العاملان بطريقتان :

إما قبل إسم المتغير مثل (++var ، --var). أو بعد إسم المتغير مثل (var++ ، var --).

var++ ; تكافئ ← var=var+1; تكافئ ← var+=1 ;

وكلها تعني أن قيمه المتغير var الجديده تساوي قيمه المتغير القديمه مضاف إليها واحد.

وبنفس المعنى إذا استخدمنا عامل النقصان. إلا أنه يعمل في الاتجاه السالب.

- ✓ سوف أشرح علي عامل الزياده فقط لتسهيل الدرس .

- عندما نكتب عامل الزياده قبل إسم المتغير نعني به أضف واحد لقيمه المتغير أولاً. وبعدها نفذ المطلوب من باقي الجمله البرمجيه.
- وعنما نكتب عامل الزياده بعد إسم المتغير نعني به نفذ المطلوب من الجمله البرمجيه أولاً وبعدها أضف واحد للمتغير.

مثلاً" إذا كان لدينا المتغير a قيمته تساوي 10 . ونسبنا قيمه (a++) للمتغير y فان قيمه y ستصبح 10 لأن عامل الزياده كتبناه بعد المتغير a . `y= a++ ;`

✓ و قيمه a تصبح 11 بعد تنفيذ هذا السطر البرمجي .

بإختصار : أسندنا قيمه المتغير a للمتغير y أولاً ثم بعد ذلك أضفنا الواحد للمتغير a.

وإذا أسندنا قيمه (++a) للمتغير y فإن قيمه y تصبح 11 لأن عامل الزياده كتبناه قبل المتغير a و قيمه a تساوي 11 عند وبعد تنفيذ السطر البرمجي. `y= ++a ;`

بإختصار أضفنا الواحد للمتغير a أولاً . ثم بعد ذلك أسندنا قيمه المتغير a للمتغير y.

والكود التالي يوضح الطريقتين السابقتين :

C:\BC5\BIN\NONAME00.CPP *

```
#include <iostream.h>
#include <conio.h>
void main()
{
int y;
int a=10;

    y=a++;
              cout<<"\n\n \t y = "<<y;
              cout<<"\n \t a = "<<a;

cout<<"\n%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%\n";

a=10;
    y=++a;
              cout<<"\n \n \t y = "<<y;
              cout<<"\n \t a = "<<a;
cout<<"\n";

getch();
}
// Mavia NorAlden
```

ومخرجات هذا الكود مايلي:

C:\BC5\BIN\NONAME00.exe

```

          y = 10
          a = 11
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


          y = 11
          a = 11
```

عندما نكتب السطر البرمجي التالي :

a++ ;

أو

++a ;

فلا فرق بينهما في هذه الحاله لأنه عند الإنتهاء من تنفيذ أي من السطرين تكون قيمه المتغير **a** الجديده تساوي قيمته القديمه مضاف إليها واحد .

❖ برنامج لإظهار الارقام من 1 الي 4 بإستخدام عامل الزياده ؟

```cpp
#include <iostream.h>
#include <conio.h>
void main()
{
int x=1;

        cout<<x<<endl;
x++;
        cout<<x<<endl;
x++;
        cout<<x<<endl;
++x;
        cout<<x<<endl;
getch();
}
```

```
C:\BC5\BIN\NON
1
2
3
4
```

- أما عامل النقصان فهو يعمل بنفس طريقه عامل الزياده إلا أنه يقوم بالطرح.

- ما ناتج مايلي :

1)

```
int  y =5 , z=10;
cout<< ( y++  +  --z )  ;
```

- بما أن قيمه المتغير y الابتدائيه تساوي 5 وعامل الزياده بعده. فإن قيمته في هذا السطر هي 5 والزياده تكون بعد الخروج من تنفيذ هذا السطر.

وبما أن قيمه المتغير z الابتدائيه 10 وعامل النقصان قبله فإن قيمته في هذا السطر هي 9 حتى بعد الخروج من تنفيذه .

إذن سيطبع علي الشاشه 14 وهو ناتج الجمع.

2)

```
int  a =10 ;
int  b = 20 ;
cout<<( --b  +  ++a ) ;
```

عامل الزياده والنقصان قبل المتغيرين إذن سيطبع علي الشاشه 30 .

3)

```
int a=3 , b=2 ;
cout<<(a++  + --b)<<endl;
cout<<(a + b);
```

✓ عندما يكون عامل الزياده أو النقصان بعد المتغير نقوم بتجاهله في هذا السطر.علماً بأن الزياده أو النقصان نتعامل معها بمجرد تنفيذ هذا السطر.

إذن سيطبع :

4
5

إنه الخلق العظيم المجتبَى

وهو مشكاة الفضيلة و المَعين

صلى الله على محمد صلى الله عليه وسلم

" لو ضربت الجبل بالزجاج ألف ضربة ما انكسر ..!! ولو سترت الصبح بكل شيء ما أنستر ..!! إنه خيار من خيار من خيار زكاه الله وكفاه .. زكى استقامته (مَا ضَلَّ صَاحِبُكُمْ وَمَا غَوَى) .. ونطقه (وَمَا يَنطِقُ عَنِ الْهَوَى) .. وعلمه (عَلَّمَهُ شَدِيدُ الْقُوَى) .. وفؤاده (مَا كَذَبَ الْفُؤَادُ مَا رَأَى) .. وبصره (مَا زَاغَ الْبَصَرُ وَمَا طَغَى) .. وصدره وذكره (أَلَمْ نَشْرَحْ لَكَ صَدْرَكَ * وَوَضَعْنَا عَنكَ وِزْرَكَ) .. وخُلقه (وَإِنَّكَ لَعَلى خُلُقٍ عَظِيمٍ) .. وزكاه كله (وَمَا أَرْسَلْنَاكَ إِلَّا رَحْمَةً لِّلْعَالَمِينَ)

فإذا كتاب الله أثنى مفصحا ** كان القصور قصارى كل فصيح "

من كتاب " إيماض البرق في خلق سيد الخلق صلى وسلم عليه الحق لفضيلة الشيخ علي بن عبدالخالق القرني"

** عوامل الـتعـيين الـحسـابـي **

هي عوامل حسابيه (+، / ، * ، % ، -) تكتب قبل علامه يساوي (=) مثل :

+=	%=	*=	- =	/=

وعندما تكتب بهذه الطريقه فإن لها معنى برمجي في لغه c++ وفي لغات أخرى أيضاً. وهي عنصر قوي في البرمجه.

وتعمل هذه العوامل مع المتغيرات وذلك لإجراء إضافه أو طرح أو قسمه أو باقي القسمه للمتغيرات .
مثلاً اذا كتبنا (; a+=10) فهذا يعني إضافه 10 للمتغير a.
أي بمعني أن قيمه المتغير a الجديده تساوي قيمته القديمه مضاف اليها الـ 10. وبهذا يمكن أن نكتب السطر البرمجي السابق بهذه الطريقه (;a= a+10) . وكل السطرين يؤدي الي نفس المعنى. أي لا فرق بينهما.

1\	int a=3; a -=2;	قيمه a الجديده ستصبح 1 .
2\	int x=100; x/=2;	قيمه المتغير x ستصبح 50 .
3\	int num=12 ; num%=3;	قيمه المتغير num تصبح صفر .(باقي القسمه)
4\	int num=12 ; num/=3;	قيمه المتغير num تصبح 4.

"يا أيها الذين آمنوا اتقوا الله حق تـقـاته و لا تموتـنّ إلا و أنتم مسلمون"

التـركيبه switch

هي أداة شرطيه تستخدم لإختبار متغير ما، له مجموعه من الحالات.أي بمعنى أخر عندما نريد أن نفعل شئ بناء علي وقوع حدث معين وفضاء العينه (مجموعه الأحداث الممكن وقوعها) لهذة الأحداث ليست بالقليله.

مثلاً إذ أردنا أن نكتب برنامج يستقبل رقم من المستخدم ويقوم البرنامج بتحويل الرقم الي صيغه لفظيه. 1 إلي one و 2 إلي tow .. وهكذا الي الرقم 10 .في هذه الحاله يمكن ان نستخدم التركيبه switch ، لسهولتها في التعامل مع مثل هذه الحالات.

الصيغه العامع لها :

```
Switch ( المتغير المراد اختباره)
        {    // بدايه التركيبه
        case   1 الحادثه  :
Statement..
break;
        case   2 الحادثه  :
Statement..
break;
        ↓
        case   n الحادثه  :
break;
        default   :
Statement..
        }   // نهايه التركيبه
```

- ✓ نكتب بعد الكلمه المحجوزه (case) أحد القيم للمتغيرالمتوقع حدوثها . ومكان Statement نكتب مانريده.
- ✓ الكلمه المحجوزه break هي لإيقاف التركيبه switch حتي لاتختبر باقي الحالات في حاله وجود الحدث المعين.
- ✓ الكلمه المحجوزه default نكتب بعدها مانريده في حاله لم تتحقق (تتطابق) أي حاله من الحالات.

بالمثال يتضح المقال : : :

❖ أكتب برنامج بإستخدام التركيبه switch لإختبار الحرف المدخل من المستخدم اذا كان y يظهر رساله ترحيب وإذا n يظهر رساله تفيد بإنتهاء عمل البرنامج :

مناقشه :

سنتعامل مع أربعه حروف لأن لغه الـ ++c حساسه لحاله الأحرف (هناك فرق بين الحرف الصغير والكبير).

عندما نريد لوقوع أكثر من حدث نفس الفعل، ما علينا إلا أن نكتب كل الحالات مع بعض ثم بعد ذلك نكتب الفعل ... لاحظ المستطيلين في الكود :

الكود :

```
#include <iostream.h>
#include <conio.h>
void main()
{
char c;
    cout<<"\n\n\t Enter ,Y, to begin or  ,n, to Exit \n";
cin>>c;

switch(c)
 {
     case 'y' :
     case 'Y' :

                cout<<"\n\n\t ******* welcome ******* \n";
           break;

    case 'n' :
    case 'N' :

                cout<<"\n\n\t **** see you soon **** \n";
           break;


    default :
                 cout<<"\n--- wrong choice ---";

 }  // end of switch
getch();
}
//Mawia NorAldeen
```

```
C:\Users\Mawia\Desktop\New folder\YES ,NO.exe

        Enter ,Y, to begin or  ,n, to Exit
 N

        **** see you soon ****
```

القراءه تعطينا المعلومات فقط ،لكن التفكير هو الذي يجعلنا نمتلك ماقرأناه

حلقــات الـتـكـرار

يوجد ثلاث أنواع من حلقات التكرار لا تختلف كثيراً عن بعضها :

1. for
2. do….. While
3. while

ونسخدم حلقات التكرار عندما نريد أن ننفذ سطر برمجي أو عدد من الأسطر البرمجيه عدد من المرات. فمثلاَ إذا أردنا أن نظهر كلمه " **Software** " ثلاثون مره علي شاشه الإخراج لابد لنا من كتابه الكلمه ثلاثون مره علي المحرر ... ولاكن بإستخدام إحدى حلقات التكرار يمكن ان نكتب الكلمه مره واحده فقط علي المحرر . وهذا ساعد من تقليل الوقوع في الاخطاء.

عندما نريد أن نكتب حلقه تكرار لابد أن نراعي لثلاثه أشياء :

أولاً : تحديد قيمه إبتدائيه أبدا منها العد وغالباً مايكون عدد صحيح مثلاً أبدا من الرقم واحد. (int i=1;).
ثانياً : شرط لتشغيل وإيقاف حلقه التكرار. كأن أقول إبدا من واحد ونفذ المطلوب مراراً وتكراراً طالما أن عداد الحلقه أقل من أو يساوي الثلاثون. (i<=30).
ثالثاً : تحديد مقدار الزياده في الحلقه أي بمعني أن الحلقه بدأت من الرقم واحد ومع كل دوره للحلقه ستزيد القيمه الابتدائيه بمقدار واحد مثلاً. (i++)

حلقه التكرار for

الصيغه العامه لها :

```
for ( قيمه ابتدائيه ; _شرط التشغيل_التكرار ; مقدار الزياده )
    {
        Statement..
    }
```

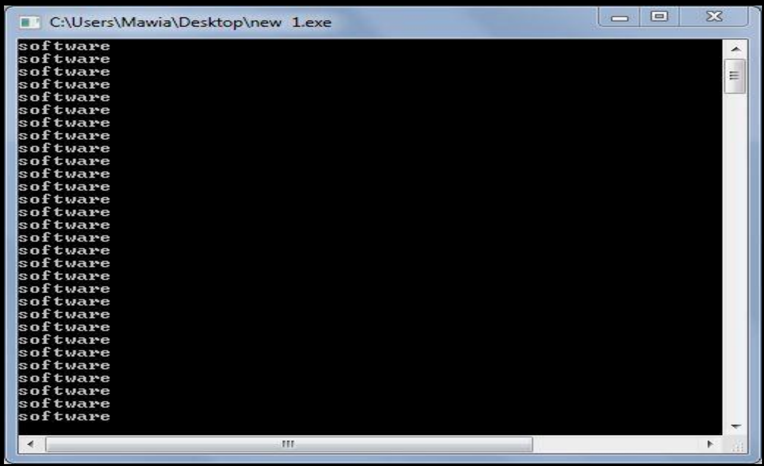
❖ لطباعة الكلمه software ثلاثون مره ستكون حلقه التكرار كالتالي:

```
for ( int i=1 ;  i<=30 ;  i++ )
{
cout<<"software \n";
}
```

وسيتم تنفيذها بالصوره التاليه : إبدا من قيمه الـ i الابتدائيه وهي واحد واذا تحقق الشرط (i<=30) نفذ العباره بين قوس البدايه والنهايه cout<<"software \n"; وإرجع للحلقه مره أخرى وسيزيد مقدار الــ i بواحد لأن i++ تعني إضافه واحد كما أوضحنا في ما سبق.ويتحقق من الشرط مره أخري وبتحققه سينفذ العباره وسيزيد قيمه الــ i ويتحقق من الشرط مره أخرى وهكذا... إلي أن يصل قيمه المتغير i لثلاثون وستنتهي الحلقه لأن الشرط لن يتحقق بعد هذه الدوره.

الكود كامل :

```
#include <iostream.h>
#include <conio.h>
void main()
{
for(int i=0; i<=30;i++)
{
cout<<"software\n";
}
getch();
}
```



✓ ملحوظه : عندما نريد تنفيذ سطر برمجي واحد بإستخدام حلقه التكرار فلن نحتاج لقوس بدايه ونهايه للحلقه لأن أول سطر يكتب بعد الحلقه يعتبر تبع الحلقه فلا حاجه لكتابه الاقواس.

✓ يمكن تعريف متغير القيمه الابتدائيه خارج الحلقة هكذا :

```
int i ;
for (i=0 ; i<=30 ; i++)
```

✓ يمكن ايقاف الحلقه والخروج منها بإستخدام أداة شرط .ويمكن أن نجعل الحلقه تعمل دون توقف.

❖ برنامج لطباعه الاعداد من 1 الي 10 بإستخدام حلقه for :

يمكننا طباعه هذه الاعداد بطباعه قيم عداد الحلقه وغالباُ مانسميه i كعُرف بين المبرمجين.

الكود :

```
#include <iostream.h>
#include <conio.h>
void main()
{
for(int i =1;i<=10;i++)
cout<<i<<"  ";
getch();

}
//Mavia NorAldeen
```



إذا لم نقم بوضع مسافه بين الاعداد فإن الأعداد ستظهر بالشكل التالي :



وهناك الكثير الكثير بما يمكن أن نفعله بهذه الحلقه ويمكن إضافه حلقه أخرى داخل الحلقه الرئيسيه وهذا ما يسمى بتداخل حلقات الـــ for .

حلقه الـتـكـرار do... while

وتعني إفعل ... طالما (بينما) أي نفذ العبارات طالما أن الشرط متحقق.وأيضاً لها قيمه إبتدائيه نبدأ منها وشرط توقف ومقدار زياده أو نقصان.

الفرق بينها وبين حلقه التكرار for : : :

ليس هناك إختلاف كبير بينهم .يمكن أن نقول بأن الإختلاف حسب إستخدامنا لها. نستخدم الـ do..while غالباً عندما لا نعرف عدد مرات التكرار اللازمه للوصول الي نتيجه معينه . مثلاً أريد برنامج يضرب اي عدد أقل من 50 في الرقم 2 ويكرر هذه العمليه إليـــن ما يصبح الرقم أكبر من أو يساوي 50 . في هذه الحاله نحن لا نعرف كم عدد مرات التكرار (الضرب في 2) المطلوبه لذلك. وهذا مثال للتقريب.ونستخدم الأخرى غالباًعندما نعلم مسبقا عدد مرات التكرار اللازمه.

الصيغه العامه :

```
; قيمه إبتدائيه لعداد الحلقه
do
     {
   statement..
   ; مقدار الزياده لعداد الحلقه
     }
while ( شرط_التشغيل_التكرار ) ;
```

- في حلقه التكرار هذه يُنفذ ما بداخل أقواس الـ do أولاً ثم يتم التحقق من الشرط الموجود داخل أقواس الكلمه المحجوزه while ويكرر ذلك إلي أن لا يتحقق شرط الاستمرار.

❖ برنامج لاستخراج الأعداد الفرديه من 10 إلي 30 بإستخدام حلقه التكرار do..while :

مناقشه :

سنبدا من الرقم 10 مروراً بكل الأرقام الصحيحه إلي أن نصل الي 30 وفي كل مره نختبر إذا كان العدد فردي أم لا .ويُطبع علي الشاشه إذا كان فردي.

تعمل الحلقه بهذه الطريقه :

إفعل { إختبر القيمه الإبتدائيه هل فرديه ؟ أم لا . إذا فرديه أرسلها لشاشه الإخراج. أضف واحد لقيمه المتغير الذي بدأنا به حلقه التكرار }

طالما : أن قيمه المتغير لم تتجاوز الثلاثون.

كود البرنامج :

```
#include <iostream.h>
#include <conio.h>
void main()
{
cout<<"\n\n ";
int i =10;
do
      {
if ( i%2 != 0)
cout<<i<<"    ";

i++;

      }
  while (i<=30);

getch();

}
//Mavia NorAldeen
```

!= تعنى لايساوي

```
C:\Users\Mawia\Desktop\new 1.exe

11   13   15   17   19   21   23   25   27   29
```

✓ يمكن كتابه هذا البرنامج بدون أداة الشرط if ولاكن على أن نستعمل عامل التعيين الحسابي لإضافه 2 مع كل دوره للحلقه. أي بمعني إبدال ;++i بـــ ;i+=2 . ولاكن يجب أن تكون القيمه

الابتدائيه 11 لأنها أول عدد صحيح فردي بعد الـ 10 وإذا بدأنا من الـ 10 سيتم طباعتها. وشرط التوقف والتشغيل يكون (i<=29) . وهذه الطريقه أسرع من الطريقه الأولي .ولكن إستخدمت الطريقه الأولي لأنها أكثر فائده للتعلم.

يصبح الكود بالشكل التالي :

```
#include <iostream.h>
#include <conio.h>
void main()
{
cout<<"\n\n ";
int i=11;
do
   {
cout<<i<<"  ";

i+=2;

   }

while (i<=29);
getch();
}
 // Mavia Noraldeen
```

سؤال : لماذا أوقفنا الحلقه عند العدد 29 ؟؟؟ جاوب أنت حبيبي الغالي.

أقل من أو يساوي	<=
أكبر من أو يساوي	>=
لا يساوي	!=



سبحان الله و الحمدلله و لا إله إلا الله و الله أكبر

حلقه التكرار while

الصيغه العامه :

قيمه إبتدائيه لعداد الحلقه ;
While (شرط التكرار)
{ // قوس البدايه
Statement..
مقدار الزياده لعداد الحلقه ;
} // قوس النهايه

هذه الحلقه لا تختلف كثيراً عن حلقه التكرار do.. while يمكن أن نقول أن الإختلاف بينهما أن هذه الحلقه (while) تختبر الشرط أولاً وإذا تحقق تبدأ بتنفيذ المطلوب. أما الثانيه فتنفذ المطلوب مره واحده علي الأقل ثم ترجع للتختبر شرط التكرار.

جدول يوضح الفرق بينهما :

No	While	do.. while
/1	تختبر الشرط أولاً ثم تبدأ في التنفيذ اذا تحقق	تنفذ ثم تختبر الشرط للاستمرار
/2	قد لا يكون لها أي تنفيذ	لها تنفيذ واحد علي الاقل
/3	لانكتب semicolon (;) نهايه تركيبه while	نكتب semicolon (;) نهايه تركيبه while

❖ برنامج لإظهار الأعداد من 100 إلي واحد بإستخدام حلقه التكرار while .

مناقشه :

ستبدأ الحلقه من الرقم100 وصولاً الي الرقم واحد وفي كل مره نقوم بإظهار عداد حلقه التكرار(i).
ولاكن مقدار الزياده سيكون بالسالب ، والأفضل أن نستخدم عامل النقصان لأن ترتيب الأرقام المطلوب تنازلي.ويكون شرط الإستمرار أن يصل عداد الحلقه الي الواحد ولا يتعداه (i>=1).

الكود :

```
#include <iostream.h>
#include <conio.h>
void main()
{
int i=100;
while (i>=1)
{

cout<<i<<"\t";
i--;
}
getch();

}
// Mawia NorAldeen
```

C:\Users\Mawia\Desktop\New folder\NONAME00.exe

```
100     99      98      97      96      95      94      93      92      91
90      89      88      87      86      85      84      83      82      81
80      79      78      77      76      75      74      73      72      71
70      69      68      67      66      65      64      63      62      61
60      59      58      57      56      55      54      53      52      51
50      49      48      47      46      45      44      43      42      41
40      39      38      37      36      35      34      33      32      31
30      29      28      27      26      25      24      23      22      21
20      19      18      17      16      15      14      13      12      11
10      9       8       7       6       5       4       3       2       1
```

الكيِّس من دان نفسه وعمل لما بعد الموت، والعاجز من أتبع نفسه هواها وتمنى على الله الأماني

بـرامج مـتـنوعه

❖ أكتب برنامج يقوم بحساب مضروب العدد بإستخدام حلقة for :

مناقشه :

من المعروف أنه لإيجاد مضروب العدد فإننا نقوم بضرب العدد في العدد الأقل منه مباشره في الأقل منه .. إلي أن نصل 1 ولايؤثر الضرب في واحد كما نعلم (محايد ضربي) .
إذا فرضنا أن الرقم المراد إيجاد مضروبه N فإن مضروبه يساوي :

N * n-1 * n-2 * ..... * 3 * 2 * 1

هذا يعني أننا سنستخدم عامل التعيين الضربي.
يجب أن يكون المتغير _المراد حفظ الناتج الأخير فيه_ قيمته الإبتدائيه واحد لأنه لا يتاثر بالضرب.

- إذن عداد الحلقه سيبدأ من الرقم المدخل من قبل المستخدم (i=n) وشرط الإستمرار أن قيمه عداد الحلقه لا يصل الصفر ويتوقف عند الواحد أو الإثنين لا يفرق. (i>=1) . ومقدار الزياده يكون بالسالب والأفضل أن نستخدم عامل النقصان لأننا نريد الوصول الي الواحد (--i)
- بما أن الصفر حاله خاصه فإننا سنخرجه من الحلقه بأداة الشرط if.

الكود:

```
#include <iostream.h>
#include <conio.h>
void main()
{
long num,f=1;
cout<<" Enter number please \n";
cin>>num;


if(num == 0)
cout<<"factorial zero = "<<1<<endl;
else
for (int i=num ;i>=1 ;i--)
f*=i;

cout<<"factorial"<<num<<"="<<f;

getch();

} // Mawia Noraldeen
```

✓ لاحظ أننا تجاهلنا التعامل مع الأرقام السالبه . ماذا سيحدث إذا أدخلنا رقم سالب ؟؟؟ أترك الاجابه لك أخي/أختي في الله.

```
C:\Users\Mawia\Desktop\...
 Enter number please
5
factorial 5 = 120
```

❖ أكتب برنامج لطباعه جداول الضرب :

مناقشه :

سنستخدم حلقتين for. وهذا مايسمي بتداخل حلقات الـ for كما سبق.

شرح سريع لها (تداخل الـ for): مع كل دوره للحلقه الخارجيه يتم تنفيذ الحلقه التي بداخلها بكاملها الي نهايتها ويرجع مره أخري الي الخارجيه ويكرر ذلك الي أن تنتهي الحلقه الخارجيه أو تلك التي كتبت اولاً أنظر الي الكود.

الكود :

```

#include <iostream.h>
#include <conio.h>
void main()
{

for(int i=1;i<=12;i++)
{

for(int j=1;j<=12;j++)

cout<<"\t"<<i<<"*"<<j<<"= "<<(i*j)<<endl;  //كل اللعبة توجد في هذا السطر

cout<<"\n\n";

}
getch();
}
//Mawia NorAldeen
```

✓ مع كل قيمه من قيم المتغير i . قيم المتغير j تتغير من 1 الي 12 .

جزء من الناتج :

```
C:\BC5\BIN\NONAME00.exe
1*1= 1
1*2= 2
1*3= 3
1*4= 4
1*5= 5
1*6= 6
1*7= 7
1*8= 8
1*9= 9
1*10= 10
1*11= 11
1*12= 12


2*1= 2
2*2= 4
2*3= 6
2*4= 8
2*5= 10
2*6= 12
2*7= 14
2*8= 16
2*9= 18
2*10= 20
2*11= 22
2*12= 24


3*1= 3
3*2= 6
3*3= 9
3*4= 12
3*5= 15
```

❖ برنامج لإظهار الأرقام الزوجيه والفرديه دون إستخدام أداة شرط. من 1 الي 100 :

مناقشه :

سنستعمل حلقتين تكرار منفصلتين عن بعضهما علي أن تكون الزياده بمقدار 2 مع كل دروه للحلقه ونبدأ بأول رقم زوجي من إتجاه الواحد لإظهار الأرقام الزوجيه و هذه الحلقه الأولى. وأول رقم فردي لإستخراج الأرقام الفرديه وتكون هذه حلقه التكرار الثانيه.

الكود :

```
Borland C++
C:\BC5\BIN\NONAME00.CPP *
#include <iostream.h>
#include <conio.h>
void main()
{

cout<<"\n\n The even number from 1 to 100 :\n\n\n";

cout<<"===========================================================================\n";
for (int i=2;i<=100;i+=2)
cout<<i<<"   ";

cout<<"\n===========================================================================";

cout<<"\n\n\n\n";



cout<<"  The odd number from 1 to 100 :\n\n\n";
cout<<"===========================================================================\n";
for (int i=1;i<=100;i+=2)
cout<<i<<"   ";
cout<<"\n===========================================================================";
getch();
}
//mavia noraldeen
```

```
C:\BC5\BIN\NONAME00.exe

        The even number from 1 to 100 :

================================================================================
2   4   6   8   10  12  14  16  18  20  22  24  26  28  30  32  34
 36  38  40  42  44  46  48  50  52  54  56  58  60  62  64  66
 68  70  72  74  76  78  80  82  84  86  88  90  92  94  96  98
 100
================================================================================


        The odd number from 1 to 100 :

================================================================================
1   3   5   7   9   11  13  15  17  19  21  23  25  27  29  31  33
35  37  39  41  43  45  47  49  51  53  55  57  59  61  63  65
67  69  71  73  75  77  79  81  83  85  87  89  91  93  95  97
99
================================================================================
```

هل يمكن أن تعمل حلقات التكرار مع الحروف ؟؟؟ لماذا لا نجرب !

❖ برنامج لطباعه الحروف الإنجليزيه الكبيره و الصغيره بإستخدام حلقه تكرار :

✓ نغير عداد الحلقه من رقمي الي حرفي .

الكود :

```
C:\BC5\BIN\NONAME00.CPP *
#include <conio.h>
#include <iostream.h>
void main()
{
 cout<<"\n\n";
for (char z='A' ;z<='Z' ; z++)
cout<<z<<" _ ";

 cout<<"\n\n\n\n";
//-------------------------------------------

for (char z='a' ;z<='z' ; z++)
cout<<z<<" _ ";

getch();
}
// Mavia NorAldeen
```

```
C:\BC5\BIN\NONAME00.exe

A _ B _ C _ D _ E _ F _ G _ H _ I _ J _ K _ L _ M _ N _ O _ P _ Q _ R _ S _ T _
U _ V _ W _ X _ Y _ Z _

a _ b _ c _ d _ e _ f _ g _ h _ i _ j _ k _ l _ m _ n _ o _ p _ q _ r _ s _ t _
u _ v _ w _ x _ y _ z _
```

❖ أكتب برنامج يقوم بإستخراج النسبه المئويه لخمس علامات. علما أن الدرجه القصوى لأي ماده مائه درجه :

الكود :

D:\NONAME00.CPP *

```
#include <conio.h>
#include <iostream.h>
void main()
{
int num,m=0;
int n;
for(int i=1;i<=5;i++)
{

cout<<" Enter mark  (" <<i<<")\n";
cin>>num;
m+=num;
}
n=(m/5);
cout<<"\t your percentage is "<<n<<"%";
getch();

}
//Mavia NorAldeen
```

D:\NONAME00.exe

```
 Enter mark  (1)
80
 Enter mark  (2)
92
 Enter mark  (3)
99
 Enter mark  (4)
70
 Enter mark  (5)
65
         your percentage is 81%
```

إذا رأيت الناس يعجبون بك ، فأعلم أنهم يعجبون بجميل أظهره الله منك ، ولا يعلمون عن قبيح ستره الله عليك ، " فاشكر الله ولاتغتر"

❖ أكتب كود لاستخراج الشكل التالي :

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

الكود :

```
#include <iostream.h>
#include <conio.h>
void main()
{

for(int q=1;q<=9; q++)
{

for(int u=1;u<=q;u++)
cout<<q;
cout<<endl;

 }
 getch();
}
//Mawia NorAldeen
```

❖ برنامج يتعامل مع نص (مصفوفه حرفيه) :

أنظر فقط !!

الكود والناتج :

```
C:\BC5\BIN\NONAME00.CPP *
#include <iostream.h>
#include <conio.h>
void main()
{
char email[15]="mawia@gmail.com";

 for(int i=0;i<15;i++)

 cout<<"\t"<<email[i]<<"\n";

 getch();

}
// Mawia NorAldeen
```

```
C:\BC5\BIN\NONA...
m
a
w
i
a
@
g
m
a
i
l
.
c
o
m
```

نصيحه:

خُذ الفكره فقط من كل برنامج (الهدف منه) وحاول أن تكتبه بطريقتك الخاصه فهذا يُنمي قدرتك في البرمجه إن شاءالله.

سؤال ؟:

_ أكتب برنامج يتضمن التركيبه switch لها ثلاث حالات مع كل حاله أكتب حلقه من حلقات التكرار الثلاثه ، وكل حلقه لها عمل مختلف من الأخرى .أي أن المستخدم عليه أن يدخل واحد من ثلاث خيارات ومع كل اختيار يظهر شئ مختلف علي شاشه الاخراج.

ملحوظه : إستخدم قوس بدايه ونهايه لكل حاله إذا كان هناك أكثر من جمله برمجيه للحاله مثل :

```
case 1 :
   {
cout<<"!!!!!!\n";
for(...;...;....)
cout<<"@@@";
  }     break;
```

عـذراَ إن قصرت و أخطأت فإني إجتهدت وقصدت الخير، وعـذراً إن نسيت فمن نفسي والشيطان وماتوفيقي إلا بالله



أسألكم الدعاء إخواني في الله

الحمدلله أولاَ وأخراَ ، وصلى الله على سيدنا محمد وعلى آله وصحبه وسلم